



**UniFly**

Jour Journey, Our Priority!

---

## **Relazione Progetto Basi Di Dati**

Autore: **Giulio Pedicone**

Matricola: **1000031068**

**A.A. 2022-2023**

# **INDICE**

<u>Descrizione del progetto</u>	<u>[1]</u>
<u>Specifiche sui dati</u>	<u>[1]</u>
<u>Glossario dei termini</u>	<u>[3]</u>
<u>Definizione delle operazioni</u>	<u>[4]</u>
<u>Progettazione concettuale</u>	<u>[5]</u>
<u>Schema scheletro</u>	<u>[5]</u>
<u>Raffinamenti</u>	<u>[6]</u>
<u>Schema finale</u>	<u>[8]</u>
<u>Vincoli non esprimibili dal diagramma E-R</u>	<u>[9]</u>
<u>Dizionario dei dati</u>	<u>[10]</u>
<u>Dizionario delle relazioni</u>	<u>[11]</u>
<u>Tabella dei Volumi</u>	<u>[12]</u>
<u>Tabella delle Frequenze</u>	<u>[13]</u>
<u>Schemi delle Operazioni</u>	<u>[14]</u>
<u>Traduzione verso il modello Relazionale</u>	<u>[19]</u>
<u>Schema Logico</u>	<u>[20]</u>
<u>Schema Fisico</u>	<u>[21]</u>
<u>Creazione Tabelle in SQL</u>	<u>[22]</u>
<u>Vista in SQL</u>	<u>[25]</u>
<u>Trigger in SQL</u>	<u>[26]</u>
<u>Query in SQL</u>	<u>[27]</u>

## **Note:**

Il presente documento è concepito esclusivamente a fini didattici, pertanto, il progetto in esso delineato non mira in alcun modo ad essere realizzato nell'ambito della pratica effettiva.

É possibile consultare la seguente repository **GitHub** per una migliore visualizzazione degli schemi sottostanti:

<https://github.com/giuliopedicone02/Database>

## Descrizione del Progetto

Il database "**UniFly**" è un sistema progettato per gestire e coordinare le attività di una **rete aeroportuale** e le operazioni delle compagnie aeree che operano al suo interno.

Il database tiene traccia di informazioni essenziali relative agli **aeroporti**, alle **compagnie aeree**, ai **voli**, ai **passengeri**, agli **aerei**, al **personale di bordo** e ai **servizi** offerti durante i voli.

Grazie a questo sistema, è possibile offrire un'esperienza di viaggio ottimizzata ai passeggeri, garantire una corretta pianificazione dei voli, gestire l'assegnazione dei **posti** e dell'equipaggio e monitorare i servizi offerti durante i viaggi.

## Specifiche sui dati

- Tabella "**Aeroporto**": Contiene i dettagli degli aeroporti, come il nome, la posizione geografica (città e paese), il codice ICAO (International Civil Aviation Organization) e un ID univoco per ciascun aeroporto. Questi dati sono essenziali per organizzare e gestire i voli in partenza e arrivo da ogni aeroporto della rete.
- Tabella "**Compagnia Aerea**": Memorizza le informazioni sulle compagnie aeree che operano nella rete aeroportuale. Ogni compagnia aerea è identificata da un ID univoco e include il nome della compagnia e il suo codice ICAO. Questi dati consentono di tenere traccia delle diverse compagnie coinvolte nelle operazioni e nelle prenotazioni.
- Tabella "**Passeggero**": Raccoglie i dettagli dei passeggeri, come il nome, il cognome, la data di nascita, l'indirizzo e il numero di telefono. Ogni passeggero è identificato da un ID univoco, il che facilita la gestione delle prenotazioni e delle informazioni personali.

- Tabella "**Volo**": Contiene tutte le informazioni relative ai voli disponibili nella rete aeroportuale. Ogni volo ha un ID univoco e include il numero del volo, la data e l'ora di partenza, i riferimenti agli aeroporti di partenza e arrivo e l'aereo che opera il volo. Questa tabella è fondamentale per la pianificazione dei voli e il monitoraggio delle operazioni aeroportuali.
- Tabella "**Aereo**": Memorizza i dettagli degli aerei che fanno parte della flotta delle compagnie aeree coinvolte. Ogni aereo è identificato da un ID univoco e contiene informazioni come il numero di serie, il modello, la capacità di passeggeri, l'anno di fabbricazione e il riferimento alla compagnia aerea proprietaria. Questi dati sono essenziali per assegnare gli aerei ai voli e garantire il rispetto delle capacità di ciascun velivolo.
- Tabella "**Posto**": Contiene i dettagli dei posti disponibili sugli aerei operanti nella rete aeroportuale. Ogni posto ha un ID univoco e include il numero del posto e la classe (Economy, Business, First). Ogni posto è collegato a un aereo specifico, consentendo una gestione accurata della disposizione dei posti sugli aerei.
- Tabella "**Prenotazione**": Gestisce le prenotazioni dei passeggeri per i voli disponibili nella rete aeroportuale. Ogni prenotazione è identificata da un ID univoco e include il codice di prenotazione generato automaticamente, la data della prenotazione, i riferimenti al passeggero, al volo prenotato e al posto selezionato. Questa tabella è fondamentale per tracciare le prenotazioni e assicurarsi che ogni passeggero abbia un posto assegnato per il volo.
- Tabella "**Equipaggio**": Memorizza i dettagli del personale di bordo, come piloti e assistenti di volo. Ogni membro dell'equipaggio ha un ID univoco e include il nome, il cognome, la data di nascita e il ruolo (ad esempio, pilota, assistente di volo). Questi dati consentono di assegnare l'equipaggio a ciascun volo e garantire una copertura adeguata di personale durante i viaggi.

- Tabella "**Servizio**": Contiene i dettagli dei servizi offerti durante i voli. Ogni servizio è identificato da un ID univoco e include una descrizione del servizio e il suo costo. Ogni servizio è collegato a un volo specifico, consentendo di gestire i servizi disponibili per i passeggeri durante i viaggi.

## Glossario dei termini

Termine	Descrizione	Sinonimi	Collegamenti
<b>Aeroporto</b>	Località di decollo ed atterraggio dei voli	Terminal	Volo
<b>Compagnia Aerea</b>	Ditta che offre i servizi di trasporto aereo	Linea Aerea	Aereo
<b>Passeggero</b>	Viaggia a bordo di un aereo	Cliente	Prenotazione
<b>Volo</b>	Operazione di trasporto aereo	Viaggio	Aeroporto, Aereo, Prenotazione, Equipaggio, Servizio
<b>Aereo</b>	Mezzo di trasporto	Velivolo	Compagnia Aerea, Volo, Posto
<b>Posto</b>	Area di un aereo designata alla seduta dei passeggeri	Poltrona	Prenotazione, Aereo
<b>Prenotazione</b>	Acquisto di un viaggio	Booking	Passeggero, Volo, Posto
<b>Equipaggio</b>	Gruppo di persone che opera e gestisce l'aereo	Staff	Volo
<b>Servizio</b>	Servizi destinati al comfort dei passeggeri	Extra	Volo

## **Definizione delle operazioni**

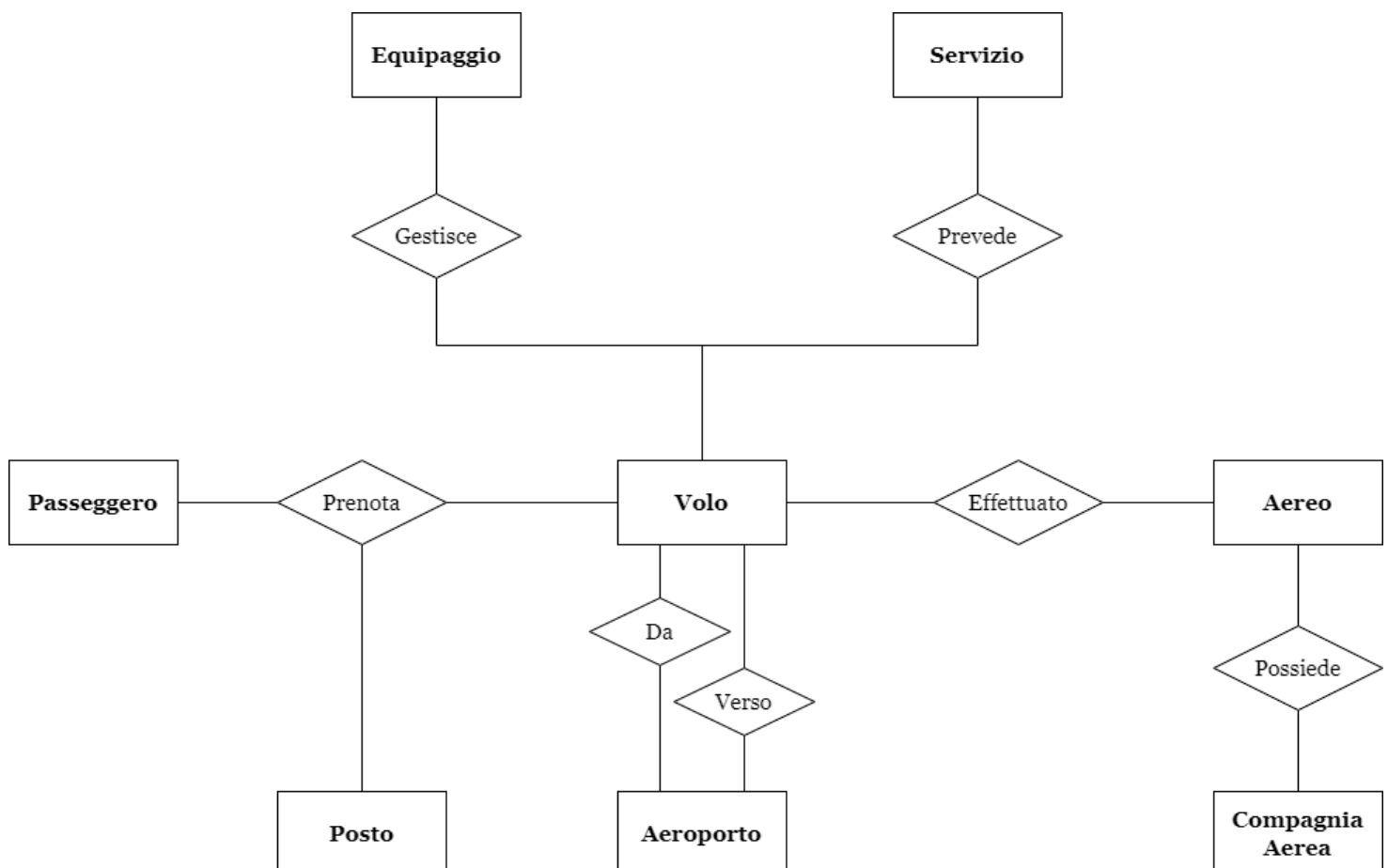
<b>Indice</b>	<b>Operazione</b>
<b>1</b>	Inserire una nuova prenotazione
<b>2</b>	Visualizzare i voli prenotati da un passeggero
<b>3</b>	Vedere il numero di aerei per ogni compagnia aerea
<b>4</b>	Trovare il volo che prevede più servizi a bordo
<b>5</b>	Ricerca di tutti i voli sotto la guida di un pilota specifico
<b>6</b>	Contare il numero di aerei per ogni produttore
<b>7</b>	Visualizzare i passeggeri che hanno prenotato un posto “Business”
<b>8</b>	Trovare la città con il maggior numero di aeroporti
<b>9</b>	Visualizzare i voli in partenza da un determinato aeroporto
<b>10</b>	Visualizzare i voli in arrivo in un determinato aeroporto

## Progettazione concettuale

Ho preso la decisione di adottare la strategia "**top-down**" per il mio progetto.

L'approccio "top-down" si basa sull'**analisi** sistematica e **dettagliata** dell'**obiettivo generale** del progetto, per poi suddividerlo in **componenti più piccole** e specifiche. In altre parole, si parte dalla **visione d'insieme** per poi scendere gradualmente nei **dettagli**. Questo permette di avere una **visione chiara** e strutturata del progetto fin dall'**inizio**, focalizzandosi sulle parti più rilevanti e rilevando eventuali problemi o sfide che potrebbero emergere durante l'implementazione.

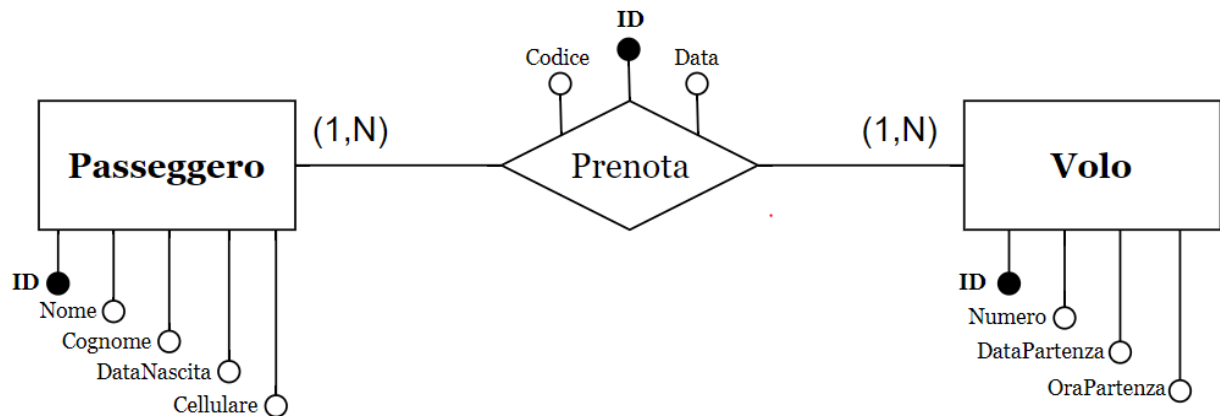
## Schema Scheletro



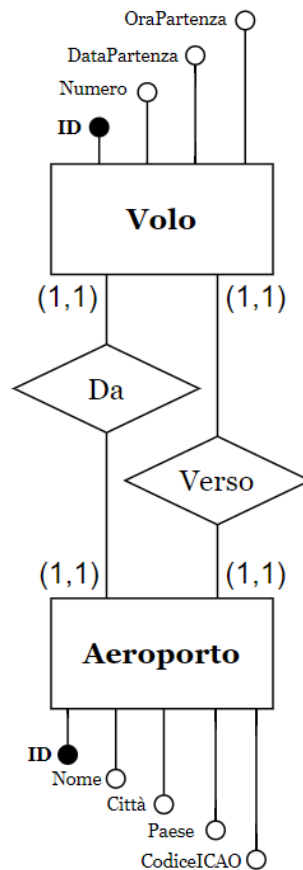
## Raffinamenti

Seguono una serie di raffinamenti dello schema scheletro sino ad arrivare ad uno schema finale e completo.

### Raffinamento n.1: Passeggero, Prenotazione, Volo

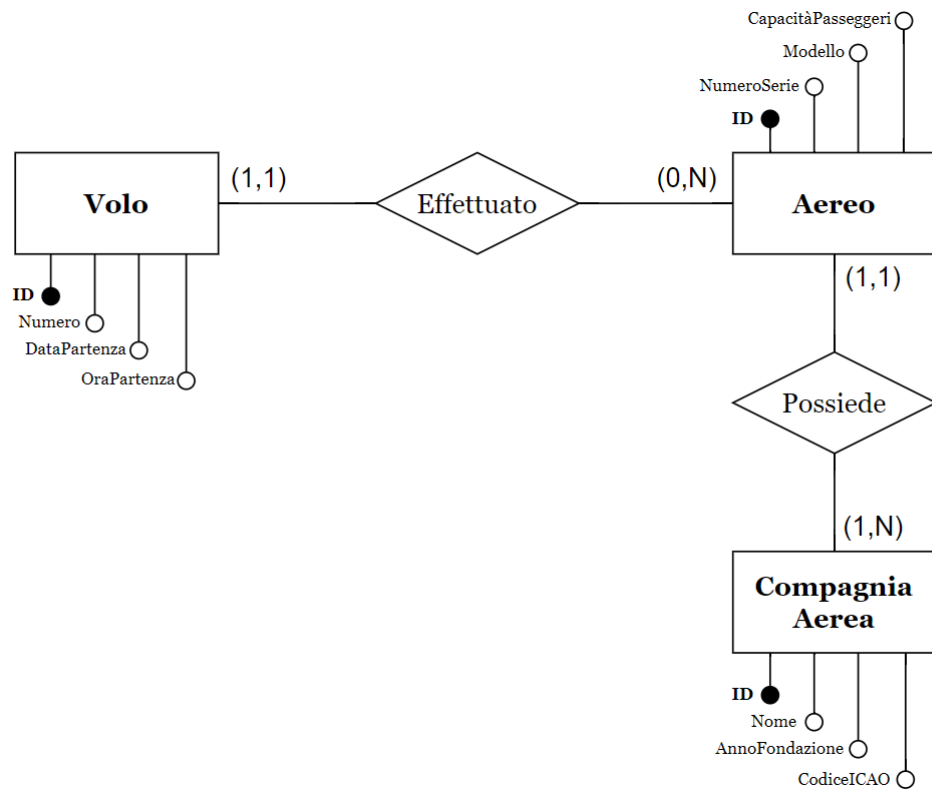


### Raffinamento n.2: Volo, Aeroporto

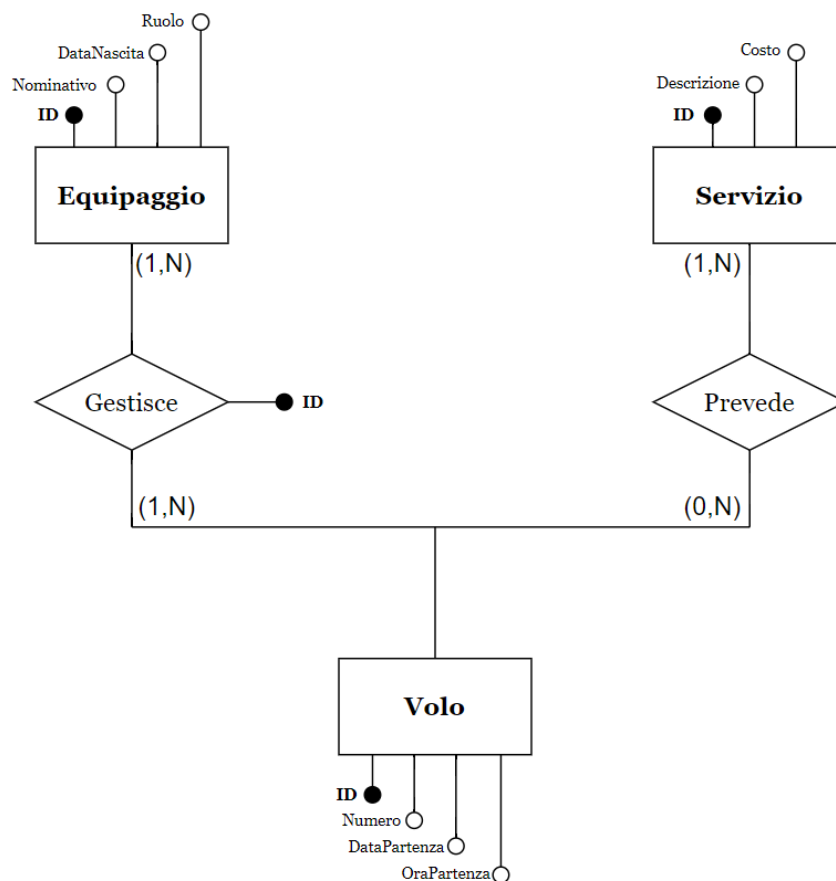




### Raffinamento n.3: Volo, Aereo, Compagnia Aerea



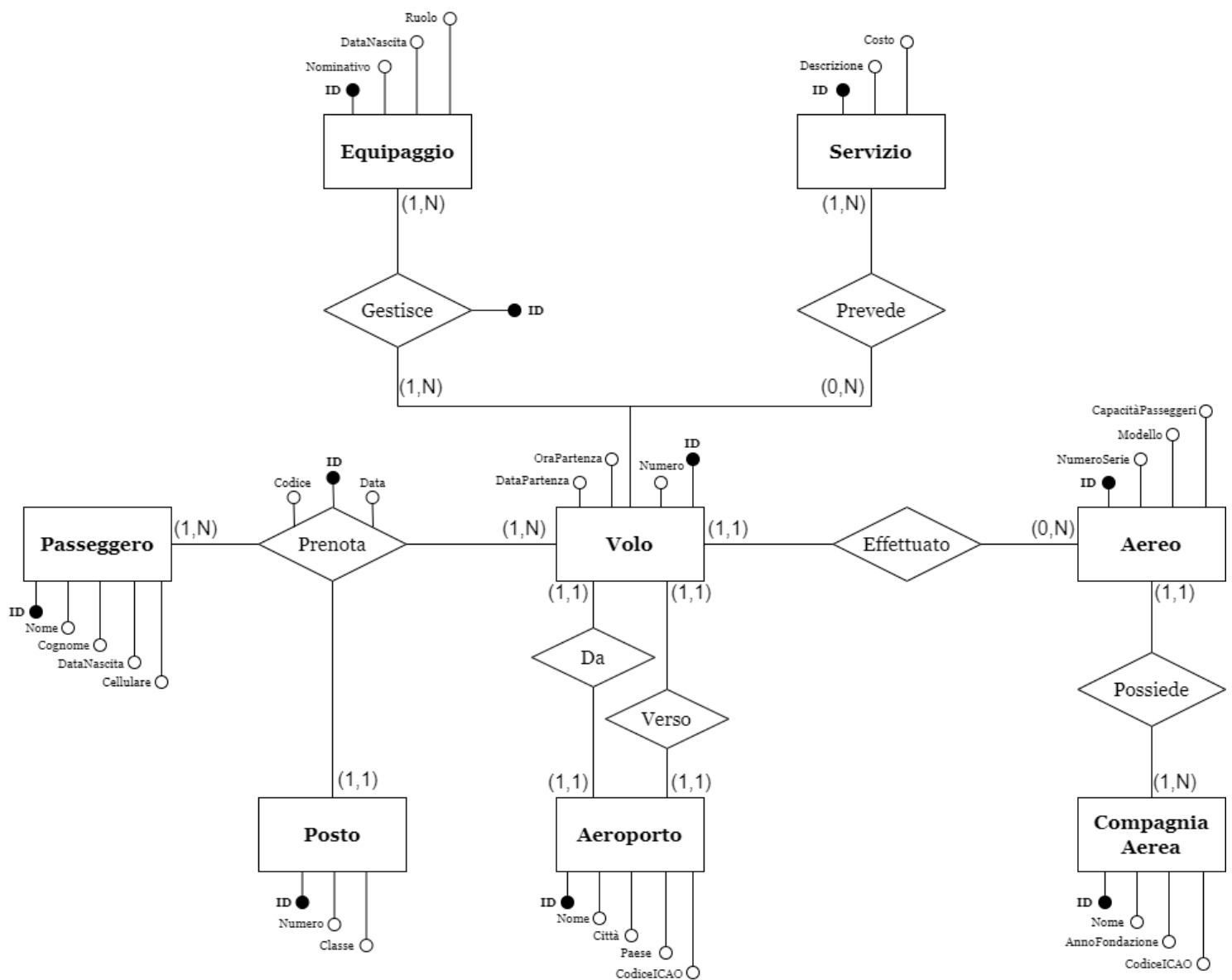
### Raffinamento n.4: Volo, Equipaggio, Servizio



## Schema Finale

Dopo aver eseguito i raffinamenti sulle cardinalità tra le tabelle presenti nel database, il risultato di tali aggiustamenti è presentato di seguito.

Si sottolinea che tali raffinamenti hanno lo scopo di fornire una **visione più precisa delle relazioni** tra le entità coinvolte nel modello di dati.



## **Vincoli non esprimibili dal diagramma E-R**

- Attributo **classe** nell'entità **Posto**: Indica la tipologia del posto selezionato in fase di prenotazione di un volo, questo attributo può assumere solo uno tra i seguenti valori:
  - Economy
  - Business
  - First
- Attributo **ruolo** nell'entità **Equipaggio**: Indica la tipologia del personale di bordo che opera presso un determinato volo, questo attributo può assumere solo uno tra i seguenti valori:
  - Pilota
  - Copilota
  - Assistente di volo

Inoltre, secondo le attuali normative in vigore, è **obbligatorio** che ogni volo sia equipaggiato con un **pilota** e un **copilota**.

## Dizionario dei dati

Entità	Descrizione	Attributi	Identificatore
<b>Aeroporto</b>	Località di decollo ed atterraggio dei voli	Nome, Città, Paese, Codice ICAO	<b>ID</b>
<b>Compagnia Aerea</b>	Ditta che offre i servizi di trasporto aereo	Nome, Anno Fondazione, Codice ICAO	<b>ID</b>
<b>Passeggero</b>	Viaggia a bordo di un aereo	Nome, Cognome, Data Nascita, Cellulare	<b>ID</b>
<b>Volo</b>	Operazione di trasporto aereo	Numero, Data Partenza, Ora Partenza	<b>ID</b>
<b>Aereo</b>	Mezzo di trasporto	Numero Serie, Modello, Capacità passeggeri	<b>ID</b>
<b>Posto</b>	Area di un aereo designata alla seduta dei passeggeri	Numero, Classe	<b>ID</b>
<b>Equipaggio</b>	Gruppo di persone che opera e gestisce l'aereo	Nome, Cognome, Data Nascita, Ruolo	<b>ID</b>
<b>Servizio</b>	Servizi destinati al comfort dei passeggeri	Descrizione, Costo	<b>ID</b>

## Dizionario delle relazioni

Associazione	Entità Partecipanti	Descrizione	Attributi
<b>Prenota</b>	Passeggero, Posto, Volo	Acquisto di un viaggio	<b>ID</b> , Codice, Data
<b>Da</b>	Volo, Aeroporto	Aeroporto in fase di decollo	-
<b>Verso</b>	Volo, Aeroporto	Aeroporto in fase di atterraggio	-
<b>Effettuato</b>	Volo, Aereo	Indica con quale aereo è stato effettuato un volo	-
<b>Possiede</b>	Aereo, Compagnia Aerea	Indica a quale compagnia aerea appartiene un aereo	-
<b>Gestisce</b>	Equipaggio, Volo	Assegna un membro dell'equipaggio ad un volo	<b>ID</b>
<b>Prevede</b>	Servizio, Volo	Indica quali servizi sono disponibili in un volo	-

## **Tabella dei Volumi**

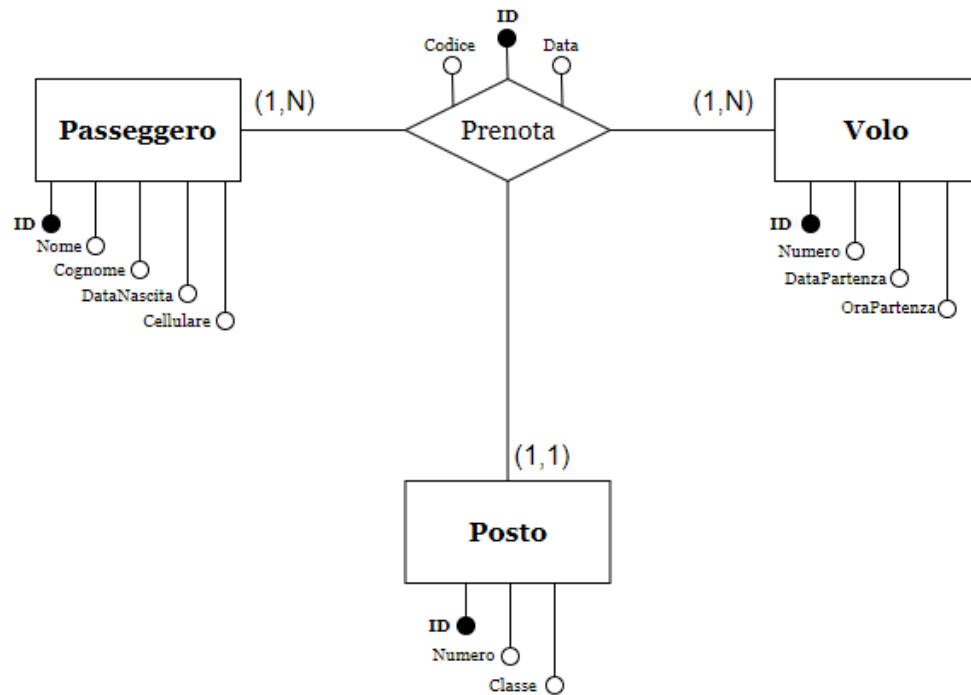
<b>Concetto</b>	<b>Tipo</b>	<b>Volume</b>
<b>Aeroporto</b>	E	50
<b>Compagnia Aerea</b>	E	10
<b>Passeggero</b>	E	500.000
<b>Volo</b>	E	100.000
<b>Aereo</b>	E	3.000
<b>Posto</b>	E	300.000
<b>Equipaggio</b>	E	2.000
<b>Servizio</b>	E	10
<b>Prenota</b>	R	40.000
<b>Da</b>	R	1
<b>Verso</b>	R	1
<b>Effettuato</b>	R	1
<b>Possiede</b>	R	1
<b>Gestisce</b>	R	2.000
<b>Prevede</b>	R	1

## **Tabella delle Frequenze**

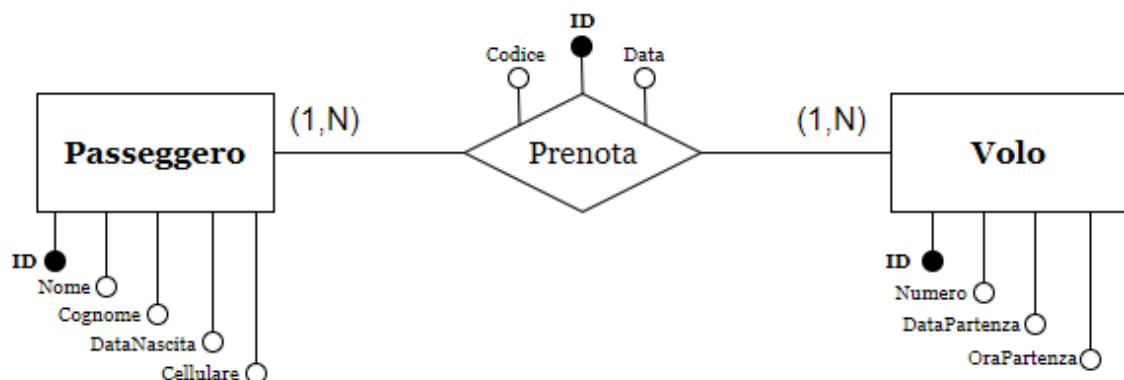
<b>Indice</b>	<b>Tipo</b>	<b>Frequenza</b>
<b>1</b>	I	10/Giorno
<b>2</b>	I	4/Mese
<b>3</b>	B	1/Anno
<b>4</b>	I	1/Mese
<b>5</b>	I	4/Settimana
<b>6</b>	B	4/Anno
<b>7</b>	I	1/Anno
<b>8</b>	I	2/Anno
<b>9</b>	I	4/Giorno
<b>10</b>	I	4/Giorno

# Schemi delle Operazioni

## 1. Inserire una nuova **prenotazione**

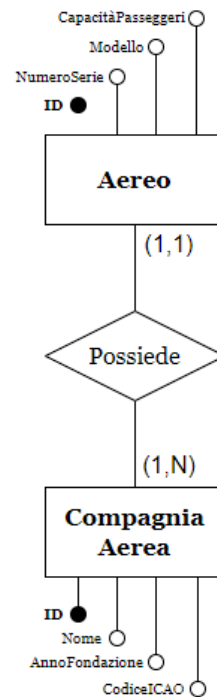


## 2. Visualizzare i **voli prenotati** da un **passeggero**

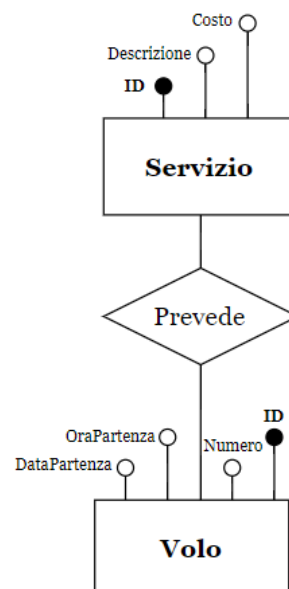




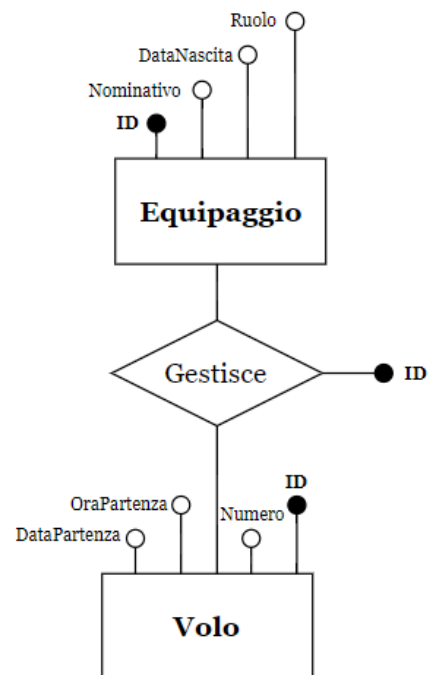
3. Vedere il numero di **aerei** per **compagnia aerea**



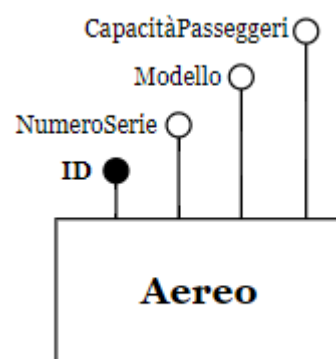
4. Trovare il **volo** che prevede più **servizi** a bordo



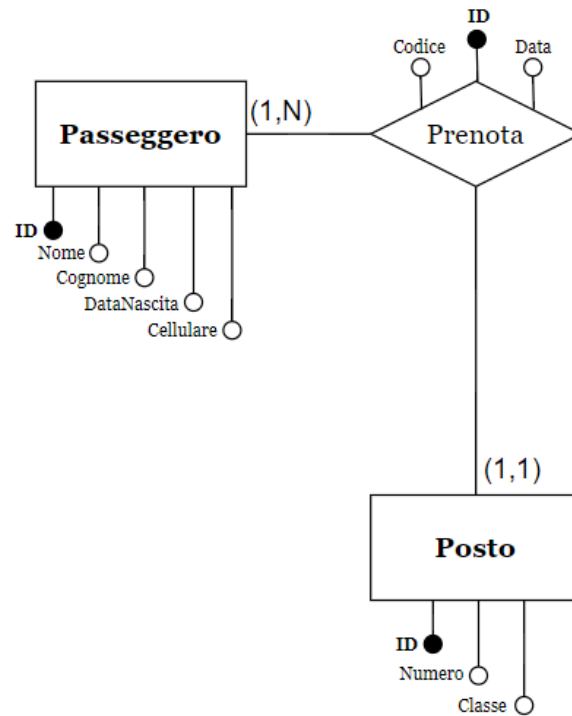
5. Ricerca di tutti i **voli** sotto la guida di un **pilota** specifico



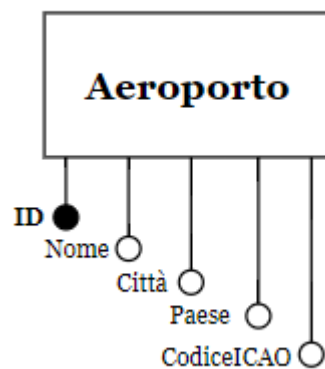
6. Contare il numero di **aerei** per ogni **produttore**



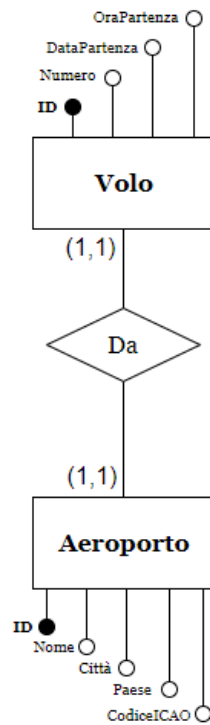
7. Visualizzare i **passaggeri** che hanno prenotato un posto **“Business”**



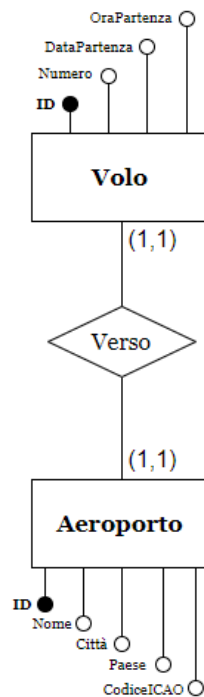
8. Trovare la città con il maggior numero di **aeroporti**



9. Visualizzare i **voli** in **partenza** da un determinato **aeroporto**



10. Visualizzare i **voli** in **arrivo** in un determinato **aeroporto**



## Traduzione verso il modello Relazionale

Per costruire un buon database, è essenziale comprendere le **relazioni** tra le diverse entità e tradurle in modo accurato.

Questo ci permette di creare uno schema logico **coerente** che **rifletta la realtà**. Solo così riusciremo a gestire i dati in modo efficace e ottenere informazioni utili dal database.

**Tradurre** correttamente le **associazioni** tra le varie entità è dunque un passaggio fondamentale per avere un **database funzionale** e di **valore**.

### Traduzione delle associazioni

- L'associazione **Prenota** viene concretizzata nell'entità **Prenotazione**.
- L'associazione **Da** viene acquisita nell'entità **Volo** sotto l'attributo "**Aeroporto\_Partenza**".
- L'associazione **Verso** viene acquisita nell'entità **Volo** sotto l'attributo "**Aeroporto\_Arrivo**".
- L'associazione **Effettuato** viene acquisita nell'entità **Volo** sotto l'attributo "**ID\_Aereo**".
- L'associazione **Possiede** viene acquisita nell'entità **Aereo** sotto l'attributo "**Compagnia**".
- L'associazione **Gestisce** viene concretizzata nell'entità **Assegnazione Equipaggio**.
- L'associazione **Prevede** viene acquisita nell'entità **Servizio** sotto l'attributo "**ID\_Volo**".

## **Schema Logico**

Lo **schema logico** è una rappresentazione dettagliata e strutturata delle tabelle, delle relazioni e dei vincoli che definiscono la struttura del database. Risulta essere dunque **essenziale** per la creazione di una base di dati ben strutturata, **coerente** e **ottimizzata**.

- **Aeroporto** (ID\_Aeroporto, Nome\_Aeroporto, Città, Paese, Codice\_ICAO)
- **CompagniaAerea** (ID\_Compagnia, Nome\_Compagnia, Anno\_Fondazione, Codice\_ICAO)
- **Passeggero** (ID\_Passeggero, Nome, Cognome, Data\_Nascita, Numero\_Telefono)
- **Aereo** (ID\_Aereo, Numero\_Serie, Modello, Capacità\_Passeggeri, Anno\_Fabbricazione, [Compagnia](#))
- **Volo** (ID\_Volo, Numero\_Volo, Data\_Partenza, Ora\_Partenza, [Aeroporto\\_Partenza](#), [Aeroporto\\_Arrivo](#), [ID\\_Aereo](#))
- **Posto** (ID\_Posto, Numero\_Posto, Classe, [ID\\_Aereo](#))
- **Equipaggio** (ID\_Equipaggio, Nome, Cognome, Data\_Nascita, Ruolo)
- **AssegnazioneEquipaggio** (ID\_AssignazioneEquipaggio , [ID\\_Equipaggio](#), [ID\\_Volo](#))
- **Servizio** (ID\_ServizioVolo, Descrizione, Costo, [ID\\_Volo](#))
- **Prenotazione** (ID\_Prenotazione, Codice\_Prenotazione, Data\_Prenotazione, [ID\\_Passeggero](#), [ID\\_Volo](#), [ID\\_Posto](#))

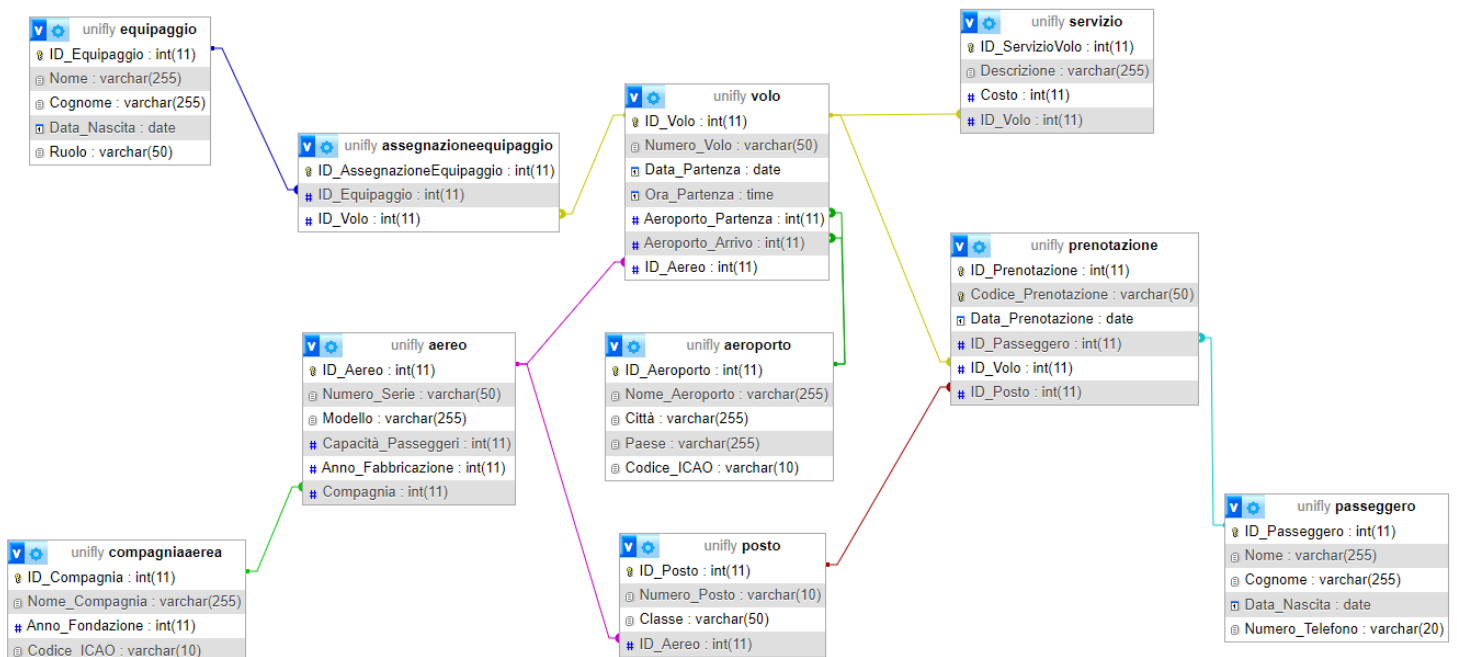
---

### **Legenda**

- NomeChiave: Chiave Primaria (PK)
- [NomeChiave](#): Chiave Esterna (FK)

# Schema Fisico

Lo **schema fisico** di un database rappresenta la struttura di memorizzazione dei dati sul **sistema di archiviazione** sottostante. Contrariamente allo schema logico, che descrive la struttura dati a un livello concettuale e indipendente dal sistema di gestione di database, lo schema fisico **definisce** come **i dati** vengono effettivamente organizzati, memorizzati e gestiti nel **DBMS**.



# Creazione delle Tabelle in SQL

## Tabella: Aeroporto

```
CREATE TABLE Aeroporto (  
    ID_Aeroporto INT PRIMARY KEY,  
    Nome_Aeroporto VARCHAR(255),  
    Città VARCHAR(255),  
    Paese VARCHAR(255),  
    Codice_ICAO VARCHAR(10)  
);
```

## Tabella: CompagniaAerea

```
CREATE TABLE CompagniaAerea (  
    ID_Compagnia INT PRIMARY KEY,  
    Nome_Compagnia VARCHAR(255),  
    Anno_Fondazione INT,  
    Codice_ICAO VARCHAR(10)  
);
```

## Tabella: Passeggero

```
CREATE TABLE Passeggero (  
    ID_Passeggero INT PRIMARY KEY,  
    Nome VARCHAR(255),  
    Cognome VARCHAR(255),  
    Data_Nascita DATE,  
    Numero_Telefono VARCHAR(20)  
);
```



### Tabella: Aereo

```
CREATE TABLE Aereo (  
    ID_Aereo INT PRIMARY KEY,  
    Numero_Serie VARCHAR(50),  
    Modello VARCHAR(255),  
    Capacità_Passeggeri INT,  
    Anno_Fabbricazione INT,  
    Compagnia INT,  
    FOREIGN KEY (Compagnia) REFERENCES CompagniaAerea (ID_Compagnia)  
);
```

### Tabella: Volo

```
CREATE TABLE Volo (  
    ID_Volo INT PRIMARY KEY,  
    Numero_Volo VARCHAR(50),  
    Data_Partenza DATE,  
    Ora_Partenza TIME,  
    Aeroporto_Partenza INT,  
    Aeroporto_Arrivo INT,  
    ID_Aereo INT,  
    FOREIGN KEY (Aeroporto_Partenza) REFERENCES Aeroporto  
(ID_Aeroporto),  
    FOREIGN KEY (Aeroporto_Arrivo) REFERENCES Aeroporto (ID_Aeroporto),  
    FOREIGN KEY (ID_Aereo) REFERENCES Aereo (ID_Aereo)  
);
```

### Tabella: Posto

```
CREATE TABLE Posto (  
    ID_Posto INT PRIMARY KEY,  
    Numero_Posto VARCHAR(10),  
    Classe VARCHAR(50),  
    ID_Aereo INT,  
    FOREIGN KEY (ID_Aereo) REFERENCES Aereo (ID_Aereo)  
);
```

### Tabella: Equipaggio

```
CREATE TABLE Equipaggio (  
    ID_Equipaggio INT PRIMARY KEY,  
    Nome VARCHAR(255),  
    Cognome VARCHAR(255),  
    Data_Nascita DATE,  
    Ruolo VARCHAR(50)  
);
```

### Tabella: Servizio

```
CREATE TABLE Servizio (  
    ID_ServizioVolo INT PRIMARY KEY,  
    Descrizione VARCHAR(255),  
    Costo DECIMAL(10, 2),  
    ID_Volo INT,  
    FOREIGN KEY (ID_Volo) REFERENCES Volo (ID_Volo)  
);
```

### Tabella: AssegnazioneEquipaggio

```
CREATE TABLE AssegnazioneEquipaggio (  
    ID_AssignazioneEquipaggio INT PRIMARY KEY,  
    ID_Equipaggio INT,  
    ID_Volo INT,  
    FOREIGN KEY (ID_Equipaggio) REFERENCES Equipaggio (ID_Equipaggio),  
    FOREIGN KEY (ID_Volo) REFERENCES Volo (ID_Volo)  
);
```

## Tabella: Prenotazione

```
CREATE TABLE Prenotazione (  
    ID_Prenotazione INT PRIMARY KEY,  
    Codice_Prenotazione VARCHAR(50) UNIQUE,  
    Data_Prenotazione DATE,  
    ID_Passeggero INT,  
    ID_Volo INT,  
    ID_Posto INT,  
    FOREIGN KEY (ID_Passeggero) REFERENCES Passeggero (ID_Passeggero),  
    FOREIGN KEY (ID_Volo) REFERENCES Volo (ID_Volo),  
    FOREIGN KEY (ID_Posto) REFERENCES Posto (ID_Posto)  
);
```

## Creazione di una vista in SQL

In SQL, una **vista** è una **tabella virtuale** che rappresenta una visualizzazione dei dati estratti da una query definita dall'utente da una o più tabelle.

Nell'esempio sottostante, si vuole visualizzare il numero di **prenotazioni** effettuate per ogni **volo** presente nel DBMS.

```
CREATE VIEW `prenotazionipervolo` AS  
SELECT ID_Volo, count(ID_Prenotazione) AS `NumeroPrenotazioni`  
FROM prenotazione  
GROUP BY ID_Volo;
```

## Creazione di un trigger SQL

UniFly, per garantire la massima tutela ai propri clienti, ha deliberato di non adottare l'**overbooking** come pratica commerciale.

Pertanto, ogni tentativo di prenotazione per un volo che risulta essere completamente prenotato sarà impedito dal **trigger** denominato **NoOverbooking**. In tali circostanze, l'utente riceverà un messaggio di errore e l'inserimento non sarà completato con successo.

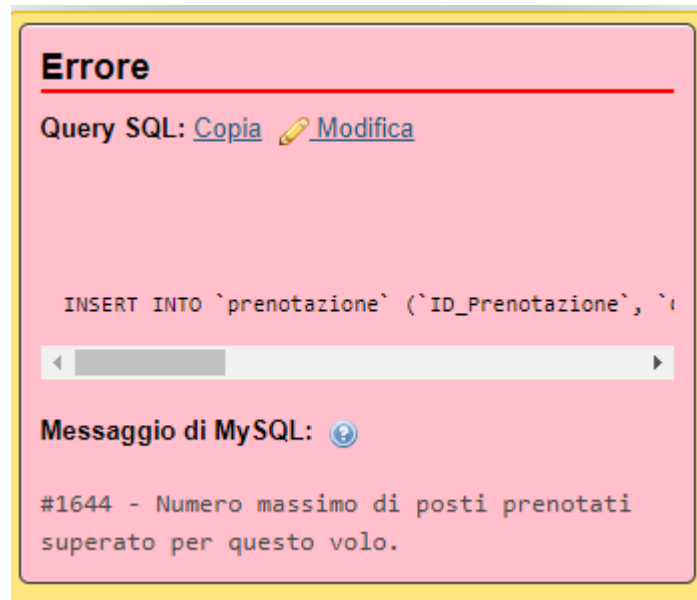
```
CREATE TRIGGER `NO_Overbooking`
BEFORE INSERT ON `prenotazione` FOR EACH ROW BEGIN
    DECLARE total_posti_prenotati INT;
    DECLARE capacita_aereo INT;

-- Calcola il numero di posti già prenotati per il volo della nuova
prenotazione
    SELECT COUNT(ID_Prenotazione) INTO total_posti_prenotati
    FROM Prenotazione
    WHERE ID_Volo = NEW.ID_Volo;

-- Ottieni la capacità massima dell'aereo del volo della nuova
prenotazione
    SELECT Capacità_Passeggeri INTO capacita_aereo
    FROM Aereo
    WHERE ID_Aereo = (
        SELECT ID_Aereo
        FROM Volo
        WHERE ID_Volo = NEW.ID_Volo
    );

-- Se il numero di posti prenotati supera la capacità dell'aereo,
genera un errore
    IF total_posti_prenotati >= capacita_aereo THEN
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'Numero massimo di posti prenotati superato
per questo volo.';
    END IF;
END
```

Nel caso in cui il trigger "**NoOverbooking**" venga attivato, verrà visualizzato un **messaggio di errore** personalizzato per notificare all'utente che l'inserimento della prenotazione non è andato a buon fine a causa dell'esaurimento dei posti disponibili sul volo desiderato. Il messaggio di errore è il seguente:



## Query in SQL

Inserire una nuova **prenotazione**:

```
INSERT INTO Prenotazione (Codice_Prenotazione, Data_Prenotazione,  
ID_Passeggero, ID_Volo, ID_Posto)  
VALUES (... , ... , [1,30], [1,30], [1,50]);
```

Per effettuare le modifiche ai valori presenti tra parentesi, è importante rispettare i **range** indicati. Si prega di prestare attenzione a non superare i limiti imposti e di mantenere la corretta coerenza dei valori.

Visualizzare i **voli prenotati** da un **passaggero**:

Questa query permette di visualizzare il numero del volo, la data e l'orario di partenza e gli aeroporti relativi alle prenotazioni effettuate da un passeggero.

```
SELECT V.ID_Volo, V.Numero_Volo, V.Data_Partenza, V.Ora_Partenza,
V.Aeroporto_Partenza, V.Aeroporto_Arrivo
FROM Volo V
JOIN Prenotazione P ON V.ID_Volo = P.ID_Volo
WHERE P.ID_Passeggero = <?>;
```

<?>: Inserisci **IDPasseggero** [1,30]

Contare il **numero** di **aerei** per ogni compagnia **aerea**:

Questa query permette di visualizzare il nome della compagnia aerea ed il numero di aerei in suo possesso.

```
SELECT CompagniaAerea.ID_Compagnia, CompagniaAerea.Nome_Compagnia,
COUNT(Aereo.ID_Aereo) AS Numero_Aerei
FROM CompagniaAerea
JOIN Aereo ON CompagniaAerea.ID_Compagnia = Aereo.Compagnia
GROUP BY CompagniaAerea.ID_Compagnia, CompagniaAerea.Nome_Compagnia;
```

Trovare il **volo** che prevede più **servizi** a bordo:

Questa query permette di visualizzare l'ID del volo che prevede il numero maggiore di servizi offerti.

```
SELECT ID_Volo, COUNT(ID_ServizioVolo) AS Numero_Servizi
FROM Servizio
GROUP BY ID_Volo
HAVING COUNT(ID_ServizioVolo) = (SELECT MAX(CountServizi)
                                FROM (SELECT COUNT(ID_ServizioVolo)
                                      AS CountServizi
                                      FROM Servizio
                                      GROUP BY ID_Volo) AS CountTable);
```

Ricerca di tutti i **voli** sotto la guida di un **pilota** specifico:

Questa query permette di visualizzare l'ID del volo, il numero del volo, la data e l'ora di partenza e gli aeroporti di riferimento.

```
SELECT V.ID_Volo, V.Numero_Volo, V.Data_Partenza, V.Ora_Partenza,  
V.Aeroporto_Partenza, V.Aeroporto_Arrivo  
FROM Volo V  
JOIN AssegnazioneEquipaggio AE ON V.ID_Volo = AE.ID_Volo  
JOIN Equipaggio E ON AE.ID_Equipaggio = E.ID_Equipaggio  
WHERE E.ID_Equipaggio = <?>;
```

<?>: Inserisci **IDEquipaggio** [1,12]

Contare il **numero** di **aerei** per ogni **produttore**:

Questa query permette di visualizzare il nome del produttore ed il numero di aerei da esso prodotti.

```
SELECT Numero_Serie as Produttore, COUNT(ID_Aereo) AS Numero_Aerei  
FROM Aereo  
GROUP BY Produttore;
```

Visualizzare i **passaggeri** che hanno prenotato un **posto** “*Business*”:

Questa query permette di visualizzare il nome ed il cognome dei passeggeri che hanno prenotato un posto di tipo Business.

```
SELECT P.ID_Passeggero, P.Nome, P.Cognome  
FROM Prenotazione PR  
JOIN Posto PO ON PR.ID_Posto = PO.ID_Posto  
JOIN Passeggero P ON PR.ID_Passeggero = P.ID_Passeggero  
WHERE PO.Classe = 'Business';
```

Trovare la **città** con il maggior numero di **aeroporti**:

Questa query permette di visualizzare il nome della città con il maggior numero di aeroporti presenti al suo interno.

```
SELECT Città, COUNT(ID_Aeroporto) AS Numero_Aeroporti
FROM Aeroporto
GROUP BY Città
ORDER BY Numero_Aeroporti DESC
LIMIT 1;
```

Visualizzare i voli in **partenza** da un determinato **aeroporto**:

Questa query permette di visualizzare il numero del volo, la data e l'orario di partenza e gli aeroporti di riferimento dei voli che partono da un determinato aeroporto.

```
SELECT V.ID_Volo, V.Numero_Volo, V.Data_Partenza, V.Ora_Partenza,
V.Aeroporto_Partenza, V.Aeroporto_Arrivo
FROM Volo V
JOIN Aeroporto AP ON V.Aeroporto_Partenza = AP.ID_Aeroporto
WHERE AP.ID_Aeroporto = <?>;
```

<?>: Inserisci **IDAeroporto** [1,10]

Visualizzare i voli in **arrivo** in un determinato **aeroporto**:

Questa query permette di visualizzare il numero del volo, la data e l'orario di partenza e gli aeroporti di riferimento dei voli che atterrano in un determinato aeroporto.

```
SELECT V.ID_Volo, V.Numero_Volo, V.Data_Partenza, V.Ora_Partenza,
V.Aeroporto_Partenza, V.Aeroporto_Arrivo
FROM Volo V
JOIN Aeroporto AP ON V.Aeroporto_Arrivo = AP.ID_Aeroporto
WHERE AP.ID_Aeroporto = <?>;
```

<?>: Inserisci **IDAeroporto** [1,10]